A SHORT-SORT M-MAX NLMS PARTIAL-UPDATE ADAPTIVE FILTER WITH APPLICATIONS TO ECHO CANCELLATION

Patrick A. Naylor and Warren Sherliker

Department of Electrical and Electronic Engineering, Imperial College London, UK

ABSTRACT

Partial-update algorithms reduce adaptive filter complexity by updating only a subset of taps at each iteration. However, they suffer a processing overhead in tap selection that can substantially reduce the computational advantages of partial-update schemes. Short-sort M-Max NLMS (SM-NLMS) addresses this problem by having the advantages of other partial-update schemes but with very low computational overhead in tap selection. SM-NLMS uses a low-complexity Short-sort procedure to perform tap selection and updates the selection periodically. We show a performance analysis based on contraction mapping for SM-NLMS using a time-varying unknown system and quantify its characteristics. Simulation results and the performance analysis show that SM-NLMS performs almost as well as NLMS but with substantially lower computational cost involved in tap selection and updating compared to other schemes. The straightforward structure and low complexity of SM-NLMS make it well suited to real-time and high-density applications such as echo cancellation and equalization.

1. INTRODUCTION

Partial-update algorithms can be suitable for adaptive filtering applications requiring real-time and/or high-density implementation [1]. Typical examples in telecommunications are echo cancellation and equalization. In such applications there is a trade-off to be made in terms of the choice of the number of taps. The adaptive filter should be long enough to model the unknown system adequately. However, shorter filters normally converge more quickly and are computationally less demanding. The use of partial-update algorithms is a good approach to this trade-off in which sufficiently long filters can be employed but only a subset of the coefficients is adapted at each iteration.

Partial-update algorithms can be seen to exploit sparseness in two ways. When the unknown system's impulse response is sparse, such as in echo cancellation for network echo and in VoIP, many of the adaptive filter's taps can be approximated to zero [2]. Alternatively, sparseness may be present in the tap update vector as a consequence of the distribution of the input samples in the $(N \times 1)$ regressor vector $\mathbf{x}_n = [x(n), x(n-1),...,x(n-N+1)]^T$. In both these cases, exploiting the sparseness properties can reduce complexity and improve performance [3].

The computational complexity of an adaptive filter comprises [4]: (i) convolution of the regressor vector $\mathbf{x}_n$ and coefficients $\mathbf{h}_n = [h_0(n), h_1(n),...,h_{N-1}(n)]^T$ at each time instant, n; (ii) filter coefficient updates; (iii) other processing overhead and housekeeping. Partial-update algorithms reduce the cost of (ii) but with an addition penalty in (iii) due to the requirement to select the sub-set of taps to update – a task involving a sort routine or other processing overhead.

This highlights the main problem we are addressing here which is that the computational saving due to the partial updating is offset (or possibly even exceeded) by the additional cost of tap selection. This motivates our study of partial-update algorithms that use efficient tap selection procedures to minimize the processing overhead. We present the Short-sort approach in the context of an NLMS-based algorithm although it can be generally applied to other adaptive filters.

Some of the first work on partial-update algorithms was done by Douglas [1] including the *periodic* and *sequential* updating schemes and Max-NLMS algorithm [1,5]. The concept was developed further by Aboulnasr leading to the M-Max NLMS algorithm and supporting convergence analysis [3,6]. A block-updating NLMS scheme was studied by Schertler [7]. More recent work has been published by Dogancay and Tanrikulu who have considered both NLMS-based schemes and Affine projection algorithms [8]. Further consideration has been given to network echo cancellation algorithms and the use of proportionate NLMS schemes [9]. Partial-update schemes have also been applied to array processing, equalization and other applications of adaptive filters.

2. SHORT-SORT M-MAX NLMS

In partial-update schemes such as M-Max NLMS, a sort procedure is employed to determine the largest amplitude samples in the regressor vector and subsequently only the taps corresponding to the largest samples are updated. In this way, algorithms of this class employ a sparse

approximation to the regressor vector and hence to the corresponding tap update vector.

The key feature of SM-NLMS is that it is able to maintain the advantages of such partial-update schemes but with a substantial reduction in the computational cost of determining the sparse approximation. This cost reduction is achieved in SM-NLMS by using the Short-sort procedure. The Short-sort operates by considering two regions of the impulse response. In Region 1, length S $(< N)$, all the taps are updated at each iteration. In Region 2, length $(N - S)$, a partial update is performed using an efficient approximation of M-Max NLMS. This approach is particularly suited to echo cancellation but also relevant to some other applications of adaptive filters including equalization. For example, in echo cancellation Region 1 corresponds approximately to the 'early reflections', which are large in amplitude and due to direct coupling and strong reflections. Region 2 corresponds generally to the decaying 'tail' of the response.

The SM-NLMS algorithm is shown in Table 1. It operates by selecting A out of S taps in Region 1 corresponding to the A largest samples in $[x(n), x(n-1), ..., x(n-S+1)]$. The tap selection is performed by an efficient insertion sort [10] that we refer to as the Short-sort as presented in Table 2. The Short-sort is used to update this tap selection every S sample periods. The tap selection tracks the largest samples in the memory of the filter as they move through Region 2. The mean number of taps adapted by SM-NLMS at each iteration is

$$L = \left(S + \frac{A}{S}(N - S) \right). \qquad (1)$$

The worst-case computation load of the Short-sort routine is $(A + A(S - A))$ comparisons per S iterations compared to $S(2\log_2(N) + 2)$ for the SORTLINE routine [11] used in M-Max NLMS.

The key assumption in SM-NLMS is that the set of A largest samples in $\mathbf{x}_n$ is a good approximation over Region 2 to the set of largest samples in $\mathbf{x}_{n+\Delta}$. The validity of this assumption depends on the properties of the input signal $\{x(n)\}$ and also on Δ $(\Delta < N)$ and will be demonstrated by simulations described below.

3. ANALYSIS

This analysis considers SM-NLMS as a combination of NLMS applied to the first S taps and an approximate M-Max NLMS updating $L = NA/S$ of the remaining taps. The analysis of NLMS is given in [4]. We describe here a new analysis of M-Max NLMS for a zero-mean Gaussian noise input.

The tap update in SM-NLMS is written

$$\mathbf{h}_{n+1} = \mathbf{h}_n + \mathbf{\Gamma}_n \mathbf{x}_n e(n) \qquad (2)$$

$$\mathbf{\Gamma}_n = \frac{2\mu}{\mathbf{x}_n^T \mathbf{x}_n} \mathbf{F}_n$$

where $\mathbf{F}_n = \text{diag}(f_0(n), f_1(n), ..., f_{N-1}(n))$ is the tap-selection matrix and $f_i(n)$ is binary and true if tap i is a selected tap at time n. We denote the misalignment vector $\mathbf{v}_n = \mathbf{h}_n - \tilde{\mathbf{h}}_n$ and the error signal $e(n) = w(n) - \mathbf{x}_n^T \mathbf{v}_n$ where $\tilde{\mathbf{h}}_n$ is the optimal system and $w(n)$ is the measurement noise. Samples corresponding to selected taps have variance κ. To introduce a time varying system model we employ a first-order Markov process [12]

$$\mathbf{h}_{n+1} = \alpha \mathbf{h}_n + \sqrt{1 - \alpha^2} \mathbf{s}_n \qquad (3)$$

$$\text{E}\left[\mathbf{h}_n \mathbf{h}_n^T \right] = \text{E}\left[\mathbf{s}_n \mathbf{s}_n^T \right] = \sigma_s^2 \mathbf{I} . \qquad (4)$$

Convergence in the mean can be derived using a contraction mapping approach [13]:

$$0 < \left\| \text{E}\left[\mathbf{I} - \mu(n) \mathbf{F}_n \mathbf{x}_n \mathbf{x}_n^T \right] \right\| < 1 \qquad (5)$$

which implies for $x(n)$ i.i.d.

$$0 < \mu(n) = \frac{2\mu}{\mathbf{x}_n^T \mathbf{x}_n} < \frac{2}{\sum_{i=0}^{N-1} f_i(n) x^2(n-i)} . \qquad (6)$$

For convergence in the mean square, we consider the trace of the autocorrelation matrix of the misalignment [13]:

$$\mathbf{v}_{n+1} = \mathbf{h}_{n+1} - \tilde{\mathbf{h}}_{n+1} = \mathbf{h}_n - \left(\alpha \tilde{\mathbf{h}}_n + \sqrt{1 - \alpha^2} \mathbf{s}_n \right) + \mathbf{\Gamma}_n \mathbf{x}_n e(n)$$

$$= \mathbf{h}_n - \tilde{\mathbf{h}}_n + \tilde{\mathbf{h}}_n - \alpha \tilde{\mathbf{h}}_n - \sqrt{1 - \alpha^2} \mathbf{s}_n + \mathbf{\Gamma}_n \mathbf{x}_n e(n)$$

$$= \left[\mathbf{I} - \mathbf{\Gamma}_n \mathbf{x}_n \mathbf{x}_n^T \right] \mathbf{v}_n + \mathbf{\Gamma}_n \mathbf{x}_n w(n) + (1 - \alpha) \tilde{\mathbf{h}}_n - \sqrt{1 - \alpha^2} \mathbf{s}_n$$

$$\mathbf{R}_{\mathbf{v},n+1} = \text{E}\left[\mathbf{v}_{n+1} \mathbf{v}_{n+1}^T \right]$$

$$= \mathbf{R}_{\mathbf{v},n} - \text{E}\left[\mathbf{\Gamma}_n \mathbf{x}_n \mathbf{x}_n^T \right] \mathbf{R}_{\mathbf{v},n} - \mathbf{R}_{\mathbf{v},n} \text{E}\left[\mathbf{x}_n \mathbf{x}_n^T \mathbf{\Gamma}_n^T \right]$$

$$+ \text{E}\left[\mathbf{\Gamma}_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n \mathbf{v}_n^T \mathbf{x}_n \mathbf{x}_n^T \mathbf{\Gamma}_n^T \right]$$

$$+ \text{E}\left[\mathbf{\Gamma}_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{\Gamma}_n^T \right] \sigma_w^2$$

$$+ (1 - \alpha)^2 \text{E}\left[\mathbf{h}_n \mathbf{h}_n^T \right] + (1 - \alpha^2) \text{E}\left[\mathbf{s}_n \mathbf{s}_n^T \right]$$

and proceed to evaluate $\text{tr}\{ \mathbf{R}_{\mathbf{v},n+1} \}$ using

$$\text{E}\left[\mathbf{\Gamma}_n \mathbf{x}_n \mathbf{x}_n^T \right] = \text{E}\left[\mathbf{x}_n \mathbf{x}_n^T \mathbf{\Gamma}_n^T \right] = \frac{L}{N} \frac{2\mu}{N \sigma_x^2} \kappa \mathbf{I}$$

$$\text{tr}\left\{ \text{E}\left[\mathbf{\Gamma}_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{v}_n \mathbf{v}_n^T \mathbf{x}_n \mathbf{x}_n^T \mathbf{\Gamma}_n^T \right] \right\} = \text{tr}\left\{ \frac{L}{N} \frac{4\mu^2}{N^2 \sigma_x^2} \kappa \left[2\mathbf{R}_{\mathbf{v},n} + \text{tr}\{ \mathbf{R}_{\mathbf{v},n} \} \right] \right\}$$

$$\text{tr}\left\{ \text{E}\left[\mathbf{\Gamma}_n \mathbf{x}_n \mathbf{x}_n^T \mathbf{\Gamma}_n^T \right] \right\} = \text{tr}\left\{ \frac{L}{N} \frac{4\mu^2}{N^2 \sigma_x^4} \kappa \mathbf{I} \right\} = \frac{4\mu^2 L}{N^2 \sigma_x^4} \kappa$$

$$\kappa = \frac{\sum_{i=0}^{N-1} x_n^2(i) f_{i,i}(n)}{\sum_{i=0}^{N-1} f_{i,i}(n)}$$

leading to

$$\mathrm{tr}\left\{\mathbf{R}_{\mathbf{v},n+1}\right\} = \mathrm{tr}\left\{\mathbf{R}_{\mathbf{v},n}\right\}\left[1 - \frac{4\mu L\kappa}{N^2\sigma_x^2} + (N+2)\frac{4\mu^2 L\kappa}{N^3\sigma_x^2}\right]$$
$$+ \frac{4\mu^2 L}{N^2\sigma_x^4}\kappa\sigma_w^2 + 2(1-\alpha)N\sigma_s^2.$$

Applying the contraction mapping concept we can show convergence when

$$\left|1 - \frac{4\mu L\kappa}{N^2\sigma_x^2}\left(1 - \frac{(N+2)\mu}{N}\right)\right| < 1.$$

which is satisfied by

$$0 < \mu < N/(N+2). \qquad (7)$$

We can then determine the excess MSE in SM-NLMS, ξ, by approximating the mean misalignment vector, η, in the steady state by the weighted sum of contributions from NLMS and M-Max NLMS:

$$\eta = \frac{S}{N}\eta_{NLMS} + \frac{N-S}{N}\eta_{MMaxNLMS} \qquad (8)$$

giving $\xi = \sigma_x^2\eta$

$$= \frac{\mu\sigma_w^2}{\phi} + \frac{(1-\alpha)N^2\sigma_s^2\sigma_x^2\left(S/N + \frac{(N-S)}{Lk}\sigma_x^2\right)}{2\mu\phi} \qquad (9)$$

where $\phi = 1 - \mu\big((N+2)/N\big)$.

4. EVALUATION

Evaluation results are presented for SM-NLMS, M-Max NLMS and NLMS for a system length of $N = 64$. SM-NLMS was tested using (A) 36 and (B) 15 tap-updates per iteration, so that $\{L, S, A, N\}$ takes values $\{36, 8, 4, 64\}$ and $\{15, 8, 1, 64\}$ respectively. M-Max NLMS was tested choosing numbers of tap-updates that result in computational costs equivalent to SM-NLMS in cases (A) and (B). The computational cost of tap selection is 14, 2.5 and 1 comparisons per iteration for M-Max NLMS, SM-NLMS(A) and SM-NLMS(B) respectively.

Mean square error (MSE) convergence tests are shown in Figure 1 using 100 trials, white Gaussian noise input, a stationary unknown system of length 64, step-size of 0.1 and measurement noise at 40 dB SNR. The convergence speed obtained for NLMS and case (A) for both partial-update methods is similar. In case (B), the convergence speed of SM-NLMS is substantially better than M-Max NLMS for the same overall complexity. Figure 2 shows SM-NLMS performance in an echo cancellation application in terms of echo return loss enhancement (ERLE) for a sentence of speech. The unknown system is a synthetic office impulse response of length 2048. Performance for SM-NLMS(A) is similar to NLMS with 64 taps. SM-NLMS(B) performs around 2 to 5 dB below NLMS during periods of speech activity, but has only one quarter the cost of tap updates.
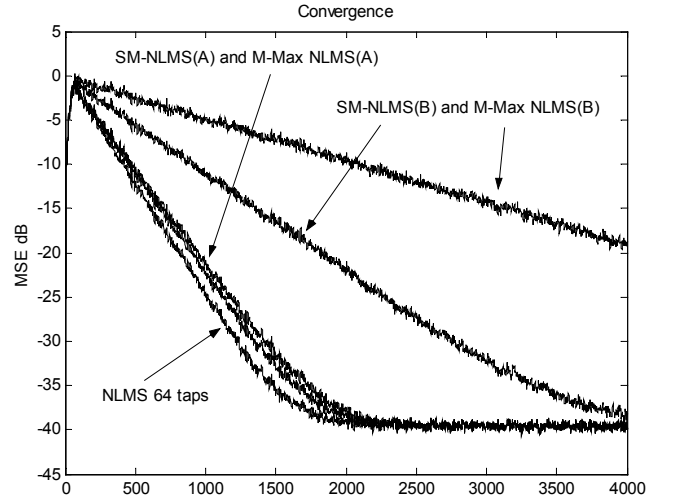

Figure 1. Convergence behavior for 40 dB SNR, Gaussian noise input, 64 tap unknown system.
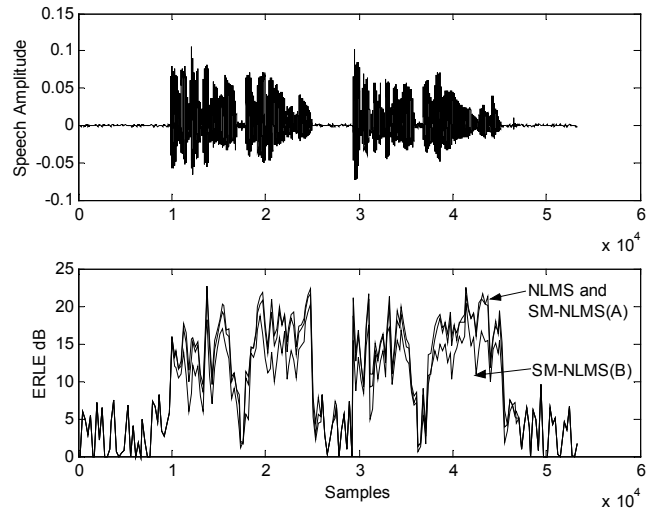

Figure 2. Echo Return Loss Enhancement for speech data

5. CONCLUSIONS

This paper has presented the SM-NLMS partial-update algorithm and its convergence analysis. The main advantage of SM-NLMS over other partial-update methods is its low complexity tap selection. Tap selection cost in SM-NLMS scales with the number of taps to be updated, unlike M-Max NLMS in which the cost depends on the filter's total taps. Performance of SM-NLMS and M-Max NLMS are roughly equal when updating more than about half the filter's taps. However, SM-NLMS out-performs M-Max NLMS when the overall complexity is reduced below this level, making SM-NLMS a good choice for high-density/low-complexity applications.

Parameters
N number of taps
μ adaptation gain
ε small positive constant
S sort window length $(< N)$
A number of selected samples $(< S)$

Signals
$\mathbf{x}_n = [x(n), x(n-1), ..., x(n-N+1)]^T$,
$d(n) = $ desired response
$\mathbf{h}_n = [h_0(n), h_1(n), ..., h_{N-1}(n)]$
$\mathbf{F}_n = \text{diag}(f_0(n), f_1(n), ..., f_{N-1}(n))$
$e(n) = $ error signal

Initialisation
$\mathbf{F}_{-1} = 0, \quad \mathbf{h}_n = 0$

Algorithm
for $n = 0,1,2,...$
$\quad \mathbf{F}_n = \text{diag}(0, f_0(n-1), f_1(n-1), ..., f_{N-2}(n-1))$
$\quad e(n) = d(n) - \mathbf{x}_n^T \mathbf{h}_n$
$\quad$ for $i = 0,1,...,S-1$

$$h_{n+1}(i) = h_n(i) + \frac{2\mu}{\mathbf{x}_n^T \mathbf{x}_n + \varepsilon} x_n(n-i)e(n)$$

$\quad$ for $i = S, S+1, ..., N-1$

$$h_{n+1}(i) = \begin{cases} h_n(i) + \dfrac{2\mu}{\mathbf{x}_n^T \mathbf{x}_n + \varepsilon} x_n(n-i)e(n), & \text{if } f_i(n) = 1 \\ h_n(i), & \text{otherwise} \end{cases}$$

$\quad$ if $(n \bmod S) = 0$
$\quad\quad$ for $i = 0,1,...,S-1$

$$f_i(n) = \begin{cases} 1, & \text{if } x(n-i) \text{ is one of the A largest values of } \mathbf{x}_n \\ 0, & \text{otherwise} \end{cases}$$

$\quad$ endif

Table 1. SM-NLMS Algorithm

REFERENCES

[1] Douglas, S. C., "Adaptive filters employing partial updates," *IEEE Trans. Circuits Syst. II*, vol. 44, no. 3, pp. 209-216, 1997.

[2] Benesty, J., Gänsler, T., Morgan, D. R., Sondhi, M. M., and Gay, S. L., *Advances in Network and Acoustic Echo Cancellation,* Springer-Verlag, 2001.

[3] Aboulnasr, T. and Mayyas, K., "Complexity reduction of the NLMS algorithm via selective coefficient update," *IEEE Trans. Signal Processing*, vol. 47, no. 5, pp. 1421-1424, 1999.

[4] Haykin, S., *Adaptive Filter Theory,* Englewood Cliffs, NJ: Prentice Hall, 1991.

[5] Douglas S. C., "Analysis and implementation of the max-NLMS adaptive filter," in *Proc. 29th Asilomar Conf. on Signals and Systems for Computing*, vol. 1, 1996, pp. 659-663.

[6] Aboulnasr T. and Mayyas, K., "MSE analysis of the M-Max NLMS adaptive algorithm," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 3, 1998, pp. 1669-1672.

[7] Schertler T., "Selective block update of NLMS type algorithms," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, vol. 3, 1998, pp. 1717-1720.

[8] Dogancay, K. and Tanrikulu, O., "Adaptive filtering algorithms with selective partial updates," *IEEE Trans. Circuits Syst. II*, vol. 48, no. 8, pp. 762-769, 2001.

[9] Tanrikulu O. and Dogancay, K., "Selective-partial-update proportionate normalized least-mean-squares algorithm for network echo cancellation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 2002, pp. 1889-1892.

[10] Knuth, D. E., *The Art of Computer Progamming, Vol. 3 Sorting and Searching,* Addison Wesley, 1973.

[11] Pitas, I., "Fast algorithms for running ordering and Max/Min calculation," *IEEE Trans. Circuits Syst.*, vol. 36, no. 6, pp. 795-804, 1989.

[12] Bershad N. J., McLaughlin, S., and Cowan, C. F. N., "Performance comparison of RLS and LMS algorithms for tracking a first order Markov communications channel," in *Proc. IEEE Int. Symposium on Circuits and Systems*, 1990, pp. 266-270.

[13] Rupp M., "Contraction mapping: an important property in adaptive filters," in *Proc. Sixth IEEE Digital Signal Processing Workshop*, 1994, pp. 273-276.

Parameters
S : sort window length $(< N)$
A : # samples to select $(< S)$
c : counter

Signals
$\mathbf{q} = [q_0, q_1, ..., q_{A-1}]$: storage
m : smallest q_i, $i = 0,1,...,A-1$
mi : index in $\mathbf{q}$ of smallest sample

Algorithm
for $n = 0,1,2,...$
$c = n \bmod S$
if $(c = 0)$ then
$\quad m = \infty$
endif
if $(c < A)$ then
$\quad q_c = c$
$\quad$ if $m > |x(n)|$ then
$\quad\quad m = |x(n)|$
$\quad\quad mi = q_c$
$\quad$ endif
else
$\quad$ if $m < |x(n)|$
$\quad\quad q_{mi} = c$
$\quad\quad m = \min$ value in $[x(n-q_i)]$, $i = 0,1,...,A-1$
$\quad\quad mi = $ value of i for $[x(n-q_i)] = m$, $i = 0,1,...,A-1$
$\quad$ endif
endif

Table 2. The Short-sort Algorithm